

MIT OpenCourseWare  
<http://ocw.mit.edu>

6.189 Multicore Programming Primer, January (IAP) 2007

Please use the following citation format:

Saman Amarasinghe and Rodric Rabbah, *6.189 Multicore Programming Primer, January (IAP) 2007*. (Massachusetts Institute of Technology: MIT OpenCourseWare). <http://ocw.mit.edu> (accessed MM DD, YYYY).  
License: Creative Commons Attribution-Noncommercial-Share Alike.

Note: Please use the actual date you accessed this material in your citation.

For more information about citing these materials or our Terms of Use, visit:  
<http://ocw.mit.edu/terms>

Name: \_\_\_\_\_

*Instructions: Write out your answers clearly in the space provided. You may use both sides of the page if necessary. Please write legibly.*

**Question 1:** In 1968, ACM published Edsger Dijkstra's letter titled *Go To Statement Considered Harmful*. Dijkstra argued that GOTO statements should be abolished because they lead to unstructured control flow and complicate the analysis and verification of programs. In 1974, Don Knuth presented an alternative viewpoint in *Structured Programming with go to Statements*. He showed that for some common programming tasks, GOTOs are the best language construct to use.

This was more than 30 years ago. Today (and looking into the future) the programming landscape is changing because of the new software crisis. How would *structured* programming help programmers more rapidly express their parallel computation? What are some of the tradeoffs they might experience?